

What is claimed is:

1. A computer program product for improving scheduling of tasks, the computer program product embodied on one or more computer readable media and comprising:

computer-readable program code means for computing whether execution of a plurality of tasks is feasible, wherein each of the tasks has an associated cost and an associated deadline;

computer-readable program code means for adding an additional amount of time to the associated cost for each of the tasks, thereby yielding a revised cost for each task, when the execution is computed to be feasible;

computer-readable program code means for iteratively repeating operation of the computer-readable program code means for computing and the computer-readable program code means for adding, until the execution is computed to be no longer feasible; and

computer-readable program code means for using the revised cost for each task as an upper limit on execution time for the task, after operation of the computer-readable program code means for iteratively repeating.

2. The computer program product according to Claim 1, wherein the additional amount of time is a fixed percentage of the associated cost for the task.

3. The computer program product according to Claim 1, wherein the additional amount of time is zero for a subset of the tasks, and for all other tasks is a fixed percentage of the associated cost for the task.

4. The computer program product according to Claim 1, wherein on a first iteration of the computer-readable program code means for adding, the additional amount of time is a fixed percentage of the associated cost for the task and wherein on other iterations, the additional amount of time is a fixed percentage of the revised cost for the task.

5. The computer program product according to Claim 1, wherein on a first iteration of the computer-readable program code means for adding, the additional amount of time is zero for a subset of the tasks, and for all other tasks is a fixed percentage of the associated cost for the task, and wherein on other iterations, the additional amount of time is a fixed percentage of the revised cost for the task.

6. The computer program product according to Claim 1, wherein the computer-readable program code means for using further comprises:

computer-readable program code means for determining, at run-time, whether a particular one of the tasks has exceeded its associated cost, and if so, computer-readable program code means for allowing the particular task to run until reaching a minimum of (1) an amount of time remaining until the task's associated deadline or (2) the upper limit on execution time for the task.

7. A system for improving scheduling of tasks, comprising:

means for computing whether execution of a plurality of tasks is feasible, wherein each of the tasks has an associated cost and an associated deadline;

means for adding an additional amount of time to the associated cost for each of the tasks,

5 thereby yielding a revised cost for each task, when the execution is computed to be feasible; and
6 means for iteratively repeating operation of the means for computing and the means for
7 adding, until the execution is computed to be no longer feasible.

1 8. The system according to Claim 7, further comprising means for using the revised cost for
2 each task as an upper limit on execution time for the task, after operation of the means for
3 iteratively repeating.

1 9. The system according to Claim 7, wherein the additional amount of time is a fixed
2 percentage of the associated cost for the task.

1 10. The system according to Claim 7, wherein the additional amount of time is zero for a
2 subset of the tasks, and for all other tasks is a fixed percentage of the associated cost for the task.

1 11. The system according to Claim 7, wherein on a first iteration of the means for adding, the
2 additional amount of time is a fixed percentage of the associated cost for the task and wherein on
3 other iterations, the additional amount of time is a fixed percentage of the revised cost for the
4 task.

1 12. The system according to Claim 7, wherein on a first iteration of the means for adding, the
2 additional amount of time is zero for a subset of the tasks, and for all other tasks is a fixed
3 percentage of the associated cost for the task, and wherein on other iterations, the additional

4 amount of time is a fixed percentage of the revised cost for the task.

1 13. The system according to Claim 7, wherein the means for using further comprises:
2 means for determining, at run-time, whether a particular one of the tasks has exceeded its
3 associated cost, and if so, means for allowing the particular task to run until reaching a minimum
4 of (1) an amount of time remaining until the task's associated deadline or (2) the upper limit on
5 execution time for the task.

1 14. A method for improving scheduling of tasks, comprising steps of:
2 computing whether execution of a plurality of tasks is feasible, wherein each of the tasks
3 has an associated cost and an associated deadline;
4 adding an additional amount of time to the associated cost for each of the tasks, thereby
5 yielding a revised cost for each task, when the execution is computed to be feasible; and
6 iteratively repeating operation of the computing step and the adding step, until the
7 execution is computed to be no longer feasible.

1 15. The method according to Claim 14, further comprising the step of using the revised cost
2 for each task as an upper limit on execution time for the task, after operation of the step of
3 iteratively repeating.

1 16. The method according to Claim 14, wherein the additional amount of time is a fixed
2 percentage of the associated cost for the task.

1 17. The method according to Claim 14, wherein the additional amount of time is zero for a
2 subset of the tasks, and for all other tasks is a fixed percentage of the associated cost for the task.

1 18. The method according to Claim 14, wherein on a first iteration of the adding step, the
2 additional amount of time is a fixed percentage of the associated cost for the task and wherein on
3 other iterations, the additional amount of time is a fixed percentage of the revised cost for the
4 task.

1 19. The method according to Claim 14, wherein on a first iteration of the adding step, the
2 additional amount of time is zero for a subset of the tasks, and for all other tasks is a fixed
3 percentage of the associated cost for the task, and wherein on other iterations, the additional
4 amount of time is a fixed percentage of the revised cost for the task.

1 20. The method according to Claim 14, wherein the using step further comprises the steps of:
2 determining, at run-time, whether a particular one of the tasks has exceeded its associated
3 cost, and if so, allowing the particular task to run until reaching a minimum of (1) an amount of
4 time remaining until the task's associated deadline or (2) the upper limit on execution time for the
5 task.